Determining the Sequence and Schedule of Job-shop Production Systems Using Genetic Algorithm by Considering Possible Values

Journal of Industrial Strategic Management
Year, 2022,
Volume 7 (Issue 1),
Pages: 42 -51.

Mostafa Mohammadi[a], *Seyed Ahmad Shayannia*[a*], *Mohamareza lotfi*[b], *Javad Rezaeian Zaidi*[a]

a Department of Industrial Management, Islamic Azad University, Firoozkooh Branch, , Firoozkooh, Iran
b Department of Industrial engineering, Islamic Azad University, Firoozkooh Branch, , Firoozkooh, Iran

Article history:
Received: 14/08/2021
Received in revised: 17/09/2021
Accepted: 21/09/2021

Abstract
Sequencing and scheduling of production in job-shop production systems is investigated in this article. Each operation specific to a job has a random duration with a mean and variance, taking into account this uncertainty in the assumptions of the model, the adaptation of the model to the real conditions of the production environment. Each job has the costs of operating the machine during processing, the cost of equipment idles for each unit of time delay in receiving the work. In this study, the optimal scheduling is costs reduction. The algorithm used to solve the problem is a genetic algorithm. The efficiency of the proposed algorithms has been tested and analyzed with a number of selected problems from the literature. This study was performed in a situation where the time of operation is uncertain following a specific statistical distribution (normal, exponential and uniform). The performance of the genetic algorithm is evaluated based on time criteria and objective function. The results obtained from the genetic algorithm were compared with the results obtained from the combined algorithm (neural network and SA algorithm) and the results obtained from the optimal solving procedures using Lingo software version 6 for 5 sample production scheduling problems. The results represent that an integrated algorithm will perform better than the genetic algorithm

Keywords: jobshop production schedule, genetic algorithm, production system.

1. Introduction

Today, with industrial development, the issue of resource constraints has become more critical, and due to the reduction of production resources, including production machines and equipment, the energy required for production and increase in operating and commissioning costs, and the unemployment of machinery, the value of system optimization resources Production is increasing. One of the most important topics attracting the attention of industry researchers, especially in recent decades, is the category of scheduling. Creating an effective and efficient programming to determine the production sequence is essentially related to increase the efficiency of production systems. This uncertainty is taken into account when processing the tasks. The problem of scheduling job-shop production is to find optimal sequence of performing various work operations related to each machine on that machine. The purpose of job-shop production scheduling is to allocate limited resources over the time to perform a group of activities and develop an appropriate schedule leading to faster access to the organization's goals. In this study, sequencing and production scheduling in job-shop production systems are examined.

Two-criterion job-shop production schedule has been investigated in this study. 2-criteria include minimizing the floating time of the operation and also minimizing the operating cost resulting from the operation of the machine and the cost incurred from the idleness of the machine. The hypotheses related to the uncertainty of the process time parameter as well as the 2-criteria in the existence of the objective function due to increasing the applicability of the problem in industry and the flexibility of the model in adapting to real conditions. However, such assumptions increase the difficulty of solving the job-shop production schedule.

[*] Corresponding Author: Tel: +989126839151 email: sheibat@yahoo.com

Since the job-shop production scheduling problem is an NP-Hard problem and one of the most difficult and important optimization problems, solving this problem indefinitely is also an NP-Hard problem. Due to the inherent complexity of job-shop production scheduling optimization problems, the use of meta-innovative methods to solve such problems improves the production of acceptable answers.

2. Review of literature

The issue of scheduling production systems of job-shop production is examined only through a probabilistic and static state in this research. Over the past 3 decades, many papers and books have been published on the timing of job-shop production systems, but only a very few have addressed the possibility of production parameters. Possible conditions for the job-shop production scheduling problem can be expressed in one or more of the following cases:

• Possibility of pre-existing and priority / delay constraints,
• Randomization of work time variables,
• Possibility of resource and sequence constraints;

Over the past decades, many algorithms have been proposed to solve the problem of traditional job-shop production such as Wang and Lu study. This paper proposes an integrated job-shop scheduling and assembly sequence planning (IJSSASP) approach for discrete manufacturing, enabling the part processing sequence and assembly sequence to be optimized simultaneously. The optimization objectives are to minimize the total production, completion time and the total inventory time of the parts during production. The interaction effects between the job shop schedule and the assembly sequence plan in discrete manufacturing are analyzed and the mathematical models including the objective functions and the constraints are established for IJSSASP. Based on the above, a non-dominated sorting genetic algorithm-II (NSGA-II) with a hybrid chromosome coding mechanism is applied to solve the IJSSASP problem (Wang, Lu, 2021) The focus of algorithms in this period has been on the problems with definite constraints on tasks. However, the existence of random processing times for such issues has been less studied. It is also rarely mentioned when the issues have possible precedence and latency constraints. In this regard, we can refer to Boyer. In his paper, he introduces a generalized flexible job-shop scheduling problem in which, besides the classical constraints of the flexible job shop scheduling problem other hard constraints such as machine capacity, time lags, holding times, and sequence-dependent setup times are taken into account. This problem is inspired by a real situation observed in a seamless rolled ring manufacturer. He proposes a mixed integer linear programming (MILP) and a constraint programming (CP) models to represent the problem. (Boyer et all,2021)

Determining delivery times for work with consideration for delay costs was also considered. The innovation of this research is based on considering the workload in determining the internal delivery time that is used to prioritize the work on the floor of the job-shop and also to determine the external delivery dates that need to determine the probability density functions of flow time. Based on the simulation results, it is shown that considering the workload in determining the delivery time creates a lower cost than the time when the workload is not considered in determining the delivery times.(Tavakkoli-Moghaddam et all,2005)

The study of scheduling issues with a focus on the probability of labor processing time is a topic that has recently been considered due to the need for flexible production systems. Therefore, in recent years, issues have been raised that have been done based on probabilistic process times with different modeling objectives. The earliest start-up time has been studied as a decision-making variable and meeting delivery time and confidence levels with different objective functions. Elmaghreby (2001) proposed a dynamic programming model for single / multi-processor processes a proposed communication to reduce the computational volume.

In 2002, a study was conducted on the production of job-shop production considering the possible processing time in the form of three distributions: normal, exponential and uniform. In assigning n work to m machine, three categories of costs have been considered:

- The cost of a fine paid in one lump sum for the delay of each work.
- Delay cost per unit time.
- Warehousing fee paid for the case of early payment or earlier delivery per unit of early payment.

The task is to determine the earliest start time to minimize the average cost of maintaining inventory and the cost of delay in delivery. In the model presented, 3 basic concepts are considered:

1) At any point in time, a choice is made based on a comparison between two tasks in allocating to a machine based on an innovative method with cost objectives.
2) A simulated model of the job-shop production problem and its combination with the proposed decision law is presented.
3) The optimization is based on the simulation performed and the decision variable is the earliest start time.

The numerical example in the simulation performed clarifies the law of decision and the optimal model is validated by repeating the simulation.Yang and Wang (Yang & Wang, 2000) examined the Near Constraint Neural Network (CSANN) with several innovative algorithms to solve the general problem of job-shop scheduling. In the proposed neural network, the ability to adjust weights and biases during processing time based on sequence constraints and resources is possible. The combination of innovative algorithms with the neural network makes it possible to improve the quality and performance of the justified response obtained from the neural network model. In this research, the simulation was performed based on four problems and through the simulation results, it was determined that the proposed neural network and the combined approaches are efficient according to the answers and the solution speed. The main point of this paper is to improve the quality of the final solution by combining the neural network model with innovative algorithms, which always leads to the creation of appropriate scheduling (optimal / near-optimal solutions).

3. The proposed mathematical model in probabilistic terms

In this part of the research, a model for scheduling production systems of job-shop production is presented in a probabilistic and static environment. In order to solve scheduling problems through the neural network, mixed and pure correct programming models have been used to show the scheduling problems of job-shop production [31, 17, 7, 5, and 32]. In this paper, a pure mathematical model is used to convert sequence constraints, resource constraints, processing start time, delivery time constraints, non-interference time constraints, and a 1% confidence interval for operation processing time into correct linear inequalities. This model has the ability to easily convert job-shop production scheduling issues to neural network design.

3.1. Model assumptions

The hypotheses considered for this model are:

- There are a good number of possible product combinations (scenarios) that can occur. Each product combination is introduced by a unique set of parts.
- The operating time of all parts on each type of machine follows a certain possible distribution.
- The types of components (combination of operations) are known in each period and are determined randomly.
- The capability and capacity of any type of machine is fixed and known over the time.
- The cost of delay for each type of part is known.
- The delivery time of each type of part is known.
- The operating cost in each type of the machine is known.
- The cost of missed opportunity is known for each car per hour.
- The number of parts, operations and machines is fixed and constant over all periods and over time.
- The limit on the number of machines must be clear and remain constant over time.
- Each type of machine can perform only one type of operation and each operation can only be performed by one machine.
- Setup times are not considered.
  Delayed and returned orders are not allowed.
- We will not have failure time for machines.
- The efficiency of the machines is 100%.
- All machines are available for using at the beginning of the course (machine installation time is zero).
- Flexibility of the machines when performing various operations is considered.

Based on the above hypotheses, the symbols, parameters and variables of problem solving are defined as follows:

3.2. Indices and definitions

$M = \{1,\ldots,m\}$: A set of machines whose m is the number of machines.

$P = \{1,\ldots,p\}$: A set of parts where p is the number of parts.

J: Symbol of the required operation of the part P

Parameters

Et_{jpm} : The average time required to process the operation of j component p on the machine m.

Vt_{jpm} : Standard deviation of the time required to process the operation of j component p on the machine m.

a_{jpm}

$\begin{Bmatrix} 1 \text{ If the operation j of part p can be processed on machine m} \\ 0\ therwise \end{Bmatrix}$

D_p : Piece delivery time p.

C_m : Operating cost of the machine m for each unit of time.

I_m Unemployment cost of car m per unit time.

O_{jpm} indicates the operation of j from the p on the machine m.

3.3. Decision variables

X_{jpms}

$\begin{Bmatrix} 1\ If\ the\ operation\ of\ j\ is\ assigned\ to\ machine\ m\ in\ sequence\ S \\ 0\ therwise \end{Bmatrix}$

Y_{jpms} : The start time of the operation of operation j fragment p on machine m in sequence s.

t_{jpm} : Optimal time required to process operation j of p on machine m with respect to α% confidence interval.

3.4.Mathematic model

$$MinZ = \sum_{m=1}^{M}\sum_{p=1}^{P}\sum_{j=1}^{Op}\left(Y_{jmax^{pms}+}X_{jmax} \times t_{jmax^{pm}} - D_p\right) + \sum_{m=1}^{M} C_m \times \left(\sum_{s=1}^{S}\sum_{p=1}^{P}\sum_{j=1}^{Op} X_{jpms} \times t_{jpms}\right) + \sum_{m=1}^{M} I_m \times \left(\sum_{s=1}^{S-1}\sum_{p=1}^{p}\sum_{j=1}^{Op} MAX\left(Y_{jpm(s+1)} - Y_{jpms} + X_{jpms} \times t_{jpm}\ .0\right)\right) \qquad (3-1)$$

3.5.Constraints

$$\sum_{m=1}^{M}\sum_{s=1}^{S}(a_{jpm} + X_{jpms} = 1 \qquad \forall j.p \qquad (3-2)$$

$$\sum_{p=1}^{P}\sum_{j=1}^{Sop} X_{jpms} \leq 1 \qquad \forall m.s \qquad (3-3)$$

$$Y_{jpms} \leq RX_{jpms} \qquad \forall j.p.m.s \qquad (3-4)$$

$$\sum_{m=1}^{M}\sum_{s=1}^{S}(Y_{jpms} + X_{jpms} * t_{jpm)} \leq \sum_{m=1}^{M}\sum_{s=1}^{S} Y_{(j+1)pms} \quad \forall j.p \qquad (3-5)$$

$$\sum_{p=1}^{P}\sum_{j=1}^{op}(Y_{jpms} + X_{jpms} * t_{jpm)} \leq \sum_{p=1}^{P}\sum_{j=1}^{op} Y_{jpm(s+1)} \quad \forall m.s \qquad (3-6)$$

$$Et_{jpm} - Z\alpha_{/2}Vt_{jpm} \leq t_{jpm} \leq Et_{jpm} + Z\alpha_{/2}Vt_{jpm} \qquad \forall j.p.m \qquad (3-7)$$

$$X \in [0.1].Y \geq 0.R \geq 0 \qquad (3-8)$$

As can be seen, the objective function of Equation (3-1) is a nonlinear integer equation that minimizes the sum of the deviations of the actual processing time from the planned value, operating costs, and opportunity cost of the machines over the planning horizon. The first statement calculates the sum of the deviation of the actual processing time from the programmed value along the programmed horizon. This sum is equal to the sum of the start time of processing operation j of p on machine m in sequence s and the optimal time required to process operation j of p on machine m. According to the confidence interval α percent if the operation j of the p-piece is assigned to the m-machine in the sequence (turn) S. The second statement calculates the operating costs of the machines. This cost is equal to the sum of the product of the number of hours required by the type of machine multiplied by the corresponding operating cost of that machine. The third statement also calculates the opportunity cost of the machines, if no parts are assigned to a machine. If the relevant machine is idle, in this case we will have a cost called the opportunity cost lost corresponding to this machine.

The constraint corresponding to Equation (3-2) ensures that each operation for each component must be assigned to only one machine and one sequence. Equation (3-3) ensures that a maximum of one operation can be assigned to each specific sequence in each machine. Equation (3-4) ensures that the processing start time is finite. Equation (3-5) ensures compliance with the sequence of operations for each component. Equation (3-6) ensures the non-interference of the processing time of the operations assigned to each machine. Equation (3-7) assumes an α percent confidence interval for the processing time of operations.

4. Computational results obtained from solving selected problems

In this part of the research, we present the computational results obtained from solving selected problems as well as their comparison and analysis. As stated earlier, the purpose of these experiments is to determine how the proposed algorithm (genetics) works in different conditions. In fact, by observing the obtained results and how the genetic algorithm works in problems of different sizes, it is possible to identify the best conditions for using this method in determining the production schedule. The answers obtained from solving the selected problems are presented in Table 2. The results show that the genetic algorithm often shows good performance in terms of time and under different conditions, and the response time is significantly reduced compared to the time required by Lingo as well as the integrated algorithm.

Table 1: In order to evaluate the proposed algorithm, five sample problems have been produced based on the literature of the subject. had. The good performance of the genetic algorithm is evident in the much lower solution time and the much smaller value of the objective function. As shown in Table 3, the difference between the values of the objective function obtained by the genetic algorithm and the Lingo software is much less than the difference between the values of the objective function obtained by the combined algorithm and the Lingo software. Also, by comparing the performance of the genetic algorithm and the combined algorithm in terms of the value of the objective function, it can be seen that the genetic algorithm has achieved a better answer in less time.

$$Gap = \frac{|Exact - Meta|}{Exact}$$

In order to ensure the effect of the size of the problem on the Figure 1: Graph comparing the values of the objective function GA and Lingo quality of the final answer, some problems were re-selected and this time solved using the selected sequence by Lingo software and it was observed that the results improved to a reasonable extent. The comparison of the obtained results is shown in the figure.According to Figure 2, it can be seen that the difference between the values of the objective function calculated by GA and Lingo is directly related to the magnitude of the problem. Also in the figure, it is shown that the values of the objective functions obtained by GA have lower (optimal) values.Figure (3) also shows the better performance of the genetic algorithm than the integrated algorithm in terms of the value of the objective function. In Figure (3), the difference between the values of the objective function is calculated by the combined algorithm and genetics.

Table 1: Computational Results of Solving Integrated Algorithm and Low Limit for Selected Problems

Test Number	Number of Parts	Number of Machine Types	Number of Operations Method	Solution Method	Number of Repetitions	Value Function Objective	Execution Time (Seconds)
1	3	2	3	GA	26	3167/6192	**124/3068**
				Hybrid algorithm	42	3811	**729/63**
				Lingo	24	2988/32	**6953**
2	5	3	3	GA	32	4039/5024	**226/4052**
				Hybrid algorithm	35	8780	**1266/8**
				Lingo	5277	3740/28	**117874**
3	6	6	6	GA	39	7240/156	**750/9534**
				Hybrid algorithm	254	20040	**4456/99**
				Lingo	**490**	**6581/96**	**162179**
4	**10**	**10**	**5**	GA	37	12009/016	**1392/203**
				Hybrid algorithm	362	29547	**8547/35**
				Lingo	576	10352/6	**190800**
5	**20**	**5**	**5**	GA	45	20809/7994	**3323/403**
				Hybrid algorithm	473	37140	**13548/94**
				Lingo	**680**	**18254/21**	**219/600**

Table 2: Computational gap between the proposed methods

The gap between the integrated algorithm and GA	The gap between the integrated algorithm and Lingo	The gap between GA and Lingo	Number of types of operations	Number of types of machines	Number of types of parts	Test number
0/203	0/215	0/0566	3	2	3	1
0/539	0/574	0/074	3	3	5	2
0/638	0/671	0/09	6	6	6	3
0/593	0/649	0/137	5	10	10	4
0/493	0/508	0/1228	5	5	20	5

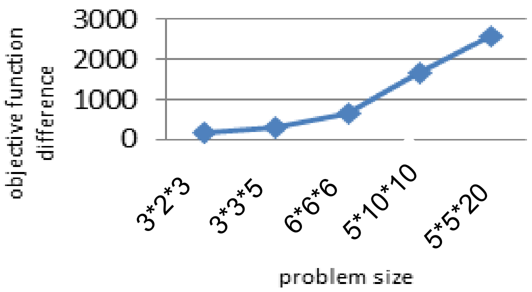

Figure 1: Graph comparing the values of the objective function GA and Lingo

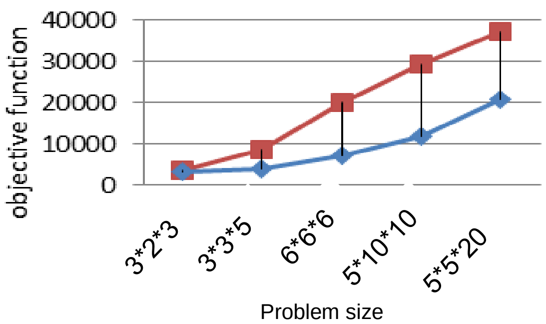

Figure 2: Comparison diagram of the value of the objective function of the combined algorithm and G

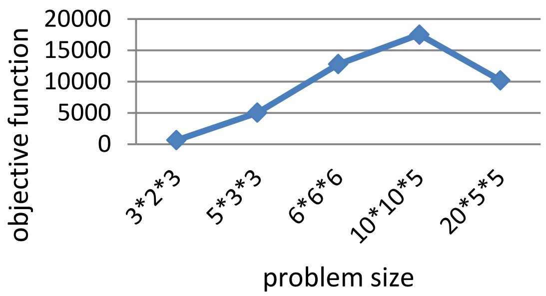

Figure 3: Diagram of the difference in the objective function values of the combined algorithm and GA

Another important factor in evaluating the performance of meta-heuristic algorithms is computational time. The figure shows the trend of increasing the solution time based on the values of the objective function GA and Lingo.

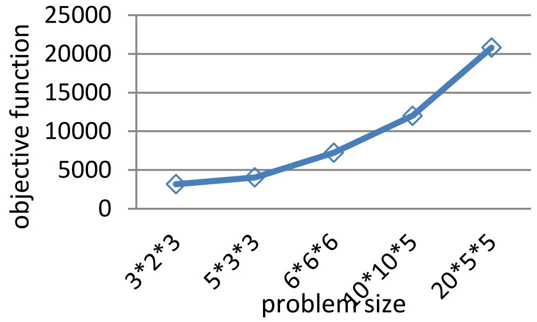

Figure 4: Graph of the process of increasing problem solving time by GA

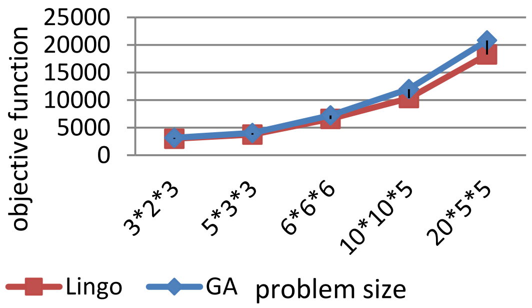

Figure 5: Trend of increasing time based on the values of the GA objective function

Genetic and Lingo algorithms are shown. According to these graphs, it can be seen that the solution time increased with increasing sample size in both methods. Also, the solving time by the genetic algorithm is significantly less than the solving time of Lingo. In Figure (4), the time of solving the sample problems by the genetic algorithm and the combined algorithm are compared and it is concluded that the genetic algorithm has performed better especially in the size of larger samples.

As shown in Figure 5, the difference in the target value based on the innovative method and Lingo software increases with the size of the problem, and this increase in the difference between the two values, due to the limitation of solving problems with large parameters by optimizers.

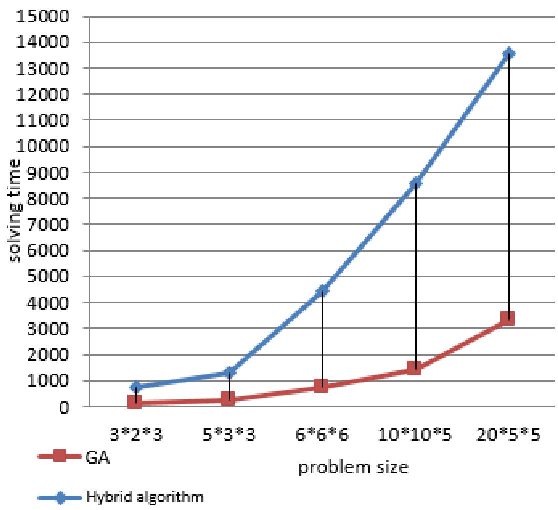

Figure 6: Comparison diagram of solution time of integrated algorithm and GA

According to the results, it can be seen that the genetic algorithm produces the near-optimal answer in less time. The reduction in the amount of calculations performed is evident in the results obtained by solving the selected numerical examples using a genetic algorithm. According to Figure (6), the difference between the solution times in the two methods increases as the problem size increases. This shows the temporal superiority of using a genetic algorithm in generating a near-optimal answer. By carefully studying the results and comparing them, the following can be seen.

The study of the results obtained from the problems solved with the combined algorithm and the genetic algorithm in this paper, shows that by increasing the problem parameters, the genetic algorithm can produce the optimal or near-optimal answer in an acceptable time. This is especially important in large issues. On the other hand, by increasing the problem parameters at the time of solution by the optimizer software, the time increase is done exponentially, which makes it practically impossible to use it for this type of problem. Comparison of the results indicates the proper performance of the genetic algorithm in creating an acceptable sequence at the right time. As a general conclusion, it can be said that the proposed algorithm in this paper provides acceptable and better answers in all cases where the parameters of large size are desired. Therefore, this algorithm is recommended for use in different types of possible problems in which the number of machines, parts and operations is high.

It is natural that because the computational time required increases exponentially as the size of the problem increases, large problems will require more computational time. By examining the results of the optimal solution, it can be seen that in problems with a large number of parameters and with increasing problem constraints, the genetic algorithm provides the optimal or near-optimal answer more easily and in less time. Since the conditions of real production systems require consideration of many constraints and parameters, the super-innovative algorithm presented in this paper (genetics) is widely used in various industries in the country.

5. Conclusion

Although research and study in determining the production schedule of the job shop has been considered by many researchers and a variety of work has been done and published in this field, but only a limited number of them to determine the production schedule of the job shop in possible circumstances. Been paid. Therefore, in this research, a genetic algorithm is proposed to determine the scheduling in probable conditions and with the aim of minimizing the discrepancy between the delivery time and the completion time of works, operating costs or unemployment of machines.

The main motivation of this research was to present a systematic method of scheduling the production of job shop production in possible conditions with the aim of reducing the discrepancy between the delivery time and the completion time of the works was also the operating costs or unemployment of the machines. In this regard, first, after reviewing and studying the work done in this field, a mathematical model was presented. Using the proposed model for real and large problems is not suitable due to computational problems and the long time it takes to solve, so the genetic algorithm was proposed to solve the problem in an acceptable time. In order to compare the proposed algorithm, five problems were randomly generated and solved using genetic algorithm. Then, the computational results obtained using the genetic algorithm were compared with the results obtained using the combined neural network and SA algorithm and analyzed. The study of these results showed good and acceptable performance of the genetic algorithm and short time to achieve the answer using the genetic algorithm in comparison with the combined neural network and SA algorithm under different conditions.

1) As the size of the problem parameters increases, the computational time by Lingo increases exponentially. This time remains almost constant in the case of large problems when using a genetic algorithm.

2) According to the obtained results, it can be seen that in case the size of the problem is small (number of parts and machines is small), the percentage difference with Lingo is equal to 5.66%, which is higher than the large size (number of parts and machines). Increases significantly and reaches 12.28%. Therefore, it seems that the size of the problem has a significant effect on the quality of the solution and as the problem grows, its computational time will increase exponentially.

3) The results show that the genetic algorithm has performed well and acceptable under different conditions and the times to reach the optimal or near-optimal solution have been significantly reduced.

4) By examining the Makespan value obtained from the solutions obtained from the two methods (genetic and combined), it was found that the solution created by the genetic algorithm is much better (less) than the Makespan value obtained from the solution obtained from the combined algorithm. Due to the importance of the Makespan parameter in scheduling most production systems, this is significant and in addition to the cost reduction goal, it is a criterion for evaluating the performance of both methods. The results indicate the proper performance of the genetic algorithm in further reducing the desired criterion in the solution created by the integrated algorithm.

5) The percentage difference of solution time in the two methods increases with increasing the size of the problem and by comparing computational times, the superiority in using the genetic method in determining the production schedule in large problems is determined.

A.J. Orman, H.P. Williams., (2005), " A Survey of Different Integer Programming Formulations of the Travelling Salesman Problem", Springer, 2-6

Aria ,Massimo and Cuccurullo ,Corrado (2017) “bibliometrix: An R-tool for comprehensive science mapping analysis”. Journal of Informetrics 11: 959–975

Alexander kinast, Karl F. Doerner, Stefanie Rinderle-Ma(2021) Biased random-key genetic algorithm for cobot assignment in an assembly/disassembly job shop scheduling problem” International Conference on Industry 4.0 and Smart Manufacturing

Calvo RW, Cordone R. (2003), “A heuristic approach to the overnight security service problem” Computers and Operations Research, 30, 1269-1287.

Carter AE, Ragsdale CT. (2002), “scheduling pre-printed newspaper advertising inserts using genetic algorithms”, Omega, 30, 415-421.

Carter AE, Ragsdale CT. (2006), “A new approach to solving multiple traveling salesman problem using genetic algorithms”, European Journal of Operational Research, 175, 246-257.

Christofides N, Mingozzi A, Toth P. (1981), “Exact algorithms for the vehicle routing problem, based on spanning tree and shortest path relaxations”, Mathematical Programming, 20,
255-282.

Dantzig GB, Fulkerson DR, Johnson SM. (1954), “Solution of a large-scale traveling salesman problem”, Operations Research, 2, 393 – 410.

Denkena, B., Wilmsmeier, S., Stock, W., 2017, Betriebsbegleitende Adaptive Arbeitsplanung und Fertigungssteuerung. VDI-Z, 159/10: 30–32.

Denkena, B., Winter, F., Wilmsmeier, S., 2019, Machine Learning in der Adaptiven Fertigungssteuerung - Genetischer Algorithmus zur Bewertung Alternativer Arbeitspläne. Fabriksoftware, 24/3: 17–20.

Deb K., Pratap A., Agarwal S., Meyarivan T., (2002), "A fast and elitist multiobjective genetic algorithm: NSGA-II ", IEEE Trans Evol Comput.

Eiben, A. E, Smith, J. E., (2003), "Introduction to Evolutionary Computing", Springer.

Fogel DB. (1990), “A parallel processing approach to a multiple traveling salesman problem using evolutionary programming”, In: Proceedings of the fourth annual symposium on parallel processing .Fullerton, CA, p, 318 – 326.

Fox BR, McMahon MB. (1991), “Genetic operators for sequencing problems”, In Foundation of Genetic Algorithms, pp, 284-300, Kaufmann.

Gavish B, Graves SC. (1978), “The traveling salesman problem and related problems”, working paper GR-078-78, Operations Research Center, Massachusetts Institute of Technology.

Gavish B, Srikanth K. (1986), “An optimal solution method for large-scale multiple traveling salesman problems”, Operations Research, 34(5), 698-717.

Gavish B. (1976), “A note on the formulation of the msalesman traveling salesman problem”, Management Science, 22(6), 704-705.

Gen, M., Cheng, R., (1997), "Genetic Algorithm and Engineering Design", Wiley Series in Engineering design and Automation New York.

Gheysens FJ. (1982), “A more compact transformation of the symmetric multiple traveling salesman problem with fixed charges”, Working paper MA/S #82-014, collage of business and Management, University of Maryland.

Gilbert KC, Hofstra RB. (1992), “A new multiperiod multiple traveling salesman problem with heuristic and application to a scheduling problem”, Decision Sciences, 23, 250-259.

Goldberg DE, Lingle RL. (1985), “Alleles, loci and the traveling salesman problem”, Proc. First Int. Conf. Genetic Algorithms and their Applications, pp, 154-159, Erlbaum.
Goldberg, D., (1989), "Genetic Algorithm in search, Optimization and machine Learning", Addison-Wesely.

Gomes LTD, Von Zuben FJ. (2002), “Multiple criteria optimization based on unsupervised learning and fuzzy inference applied to the vehicle routing problem”, Journal of intelligent and Fuzzy Systems, 13, 143 – 154.

Gromicho J, Paixão J, Branco I. (1992), “Exact solution of multiple traveling salesman problems”, Combinatorial optimization, NATO ASI Series, vol. F82. Berlin, Springer, p, 291- 292.

GuoXing Y. (1995), “Transformation of multidepot multisalesmen problem to the standard traveling salesman problem”, European Journal of Operational Research, 81, 557 – 560.

Hadjionstantinou E, Roberts D. (2002) “Routing under uncertainty: an application in the scheduling of field service engineers”, SIAM Monographs on discrete Mathematics and Applications, Philadelphia, P, 331- 352.
Hong S, Padberg MW. (1977), “A note on the symmetric multiple traveling salesman problem with fixed charges”, Operations Research, 25(5), 871 – 874.

Hsu C, Tsai M, Chen W. (1991), “A study of feature-mapped approach to the multiple traveling salesmen problem”, IEEE International Symposium on circuits and Systems, 3, 1589 – 1592.

Javadian N, Adbesh F, Kardel K. (2007), "A new approach to solving the fixed destination multi-depot multiple traveling salesman problem using genetic algorithms”, Proceeding to seventh Int. Conf. on Optimization: Techniques and Applications (ICOTA7), Cobe, Japan.

Vincent,Boyer,Jobish Vallikavungal, Xavier Cantú Rodríguez,M. Angélica Salazar-Aguilara(2021)” The generalized flexible job shop scheduling problem” Computers & Industrial Engineering Volume 160, October 2021, 107542

Zi-Yue Wang,CongLu(2021)” An integrated job shop scheduling and assembly sequence planning approach for discrete manufacturing, Journal of Manufacturing Systems Volume 61, October 2021, Pages 27-44

Joines JA, Culbreth CT. (1996), “Manufacturing cell designe: An integer programming model employing genetic”, IIE Transactions, 28(1), 69-85.

Junker R, Volgenant T. (1988), “An improved transformation of the symmetric multiple traveling salesman problem”. Operations Research, 36(1), 163 – 167.

Kara I, Bektas T. (2006), “Integer programming formulations of multiple salesman problems and its variations”, European Journal of Operational Research, 174, 1449-1458.

Kim KH, Park Y. (2004), “A crane scheduling method for port container terminals”. European Journal of Operational Research, 156, 752-768.

Kulkarni RV, Bhave PR. (1985), “Integer programming formulations of vehicle routing problems”, European Journal of Operational Research, 20, 58- 67.

Laporte G, Nobert Y, Desrochers M. (1985), “Optimal routing under capacity and distance restrictions”.

Operations Research, 33(5), 1050-1073.

Laporte G, Nobert Y, Taillefer S. (1988), “Solving a family of multi-depot vehicle routing and location-routing problems”, Transportation Science, 22(3), 161 – 172.

Laporte G, Nobert Y. (1980), “A cutting planes algorithm for the m-salesmen problem”, Journal of the Operational Research Society, 31, 1017-1023.

Laporte G. (1992), “The vehicle routing problem: an overview of exact and approximate algorithms”, European Journal of Operational Research, 59, 345-358.

Lenstra JK Rinnooy Kan AHG. (1975), “Some simple applications of the traveling salesman problem”, Operational Research Quarterly, 26, 717-733.

Lin S, Kernighan B.(1973),“An effective heuristic algorithm for the traveling salesman problem”, Operation Research, 21 , 498 – 516.

Malmborg C.(1996), “A genetic algorithm for service level based vehicle scheduling”, European Journal of Operational Research, 93(1), 121-134.

Melanie, M., (1999), "An Introduction to Genetic Algorithms", A Bradford Book The MIT Press Cambridge, Massachusetts-London, England, Fifth Printing.

Miller CE, Tucker AW, Zemlin RA. (1960), “Integer programming formulation of traveling salesman problems”, Journal of Association for Computing Machinery, 7, 326 -329.

Mitrovi´ c-Mini´c S, Krishnamurti R, Laporte G.(2004) “Double-horizon based heuristics for the dynamic pickup and delivery problem with time windows”, Transportation Research, 28(8), 669-685.

Modares A, Somhom S, Enkawa T. (1999), “A self-organizing neural network approach for multiple traveling salesman and vehicle routing problems”. International Transactions in Operational Research, 6, 591 – 606.

Mole RH, Johnson DG, Wells K. (1983), “Combinatorial analysis for route first-cluster second vehicle routing”, Omega, 11(5), 507-512.

Okonjo-Adigwe C. (1988), “An effective method of balancing the workload amongst salesmen”, Omega, 16(2), 159-163.

Oliver IM, Smith DJ, Holland RC. (1987), “A study of permutation crossover operators on the traveling salesman problem”, Proc. second Int. Conf. Genetic Algorithms and their Application, pp, 224-230, Erlbaum.

Orloff CS. (1974), “Routing a fleet of M vehicles to/from a central facility”, Networks, 4, 147-162.

Park YB. (2001), “A hybrid genetic algorithm for the vehicle scheduling problem with due times and time deadlines”, International Journal of Productions Economics, 73(2), 175-188.

Poon PW, Carter JN. (1995), “Genetic algorithm crossover operators for ordering applications”, Computers operations researches, 22, 135-147.

Potvin J, Lapalme G, Rousseau J. (1989), “A generalized k-opt exchange procedure for the MTSP” INFOR, 27(4), 474 – 481.

Rao MR. (1980), “A note on the multiple traveling salesman problem”, Operations Research, 28(3), 628 – 632.

Ruland KS, Rodin EY. (1997) “The pickup and delivery problem”, Computers and Mathematics with Applications, 33(12), 1-13.

Russell RA. (1977), “An effective heuristic for the m-tour traveling salesman problem with some side conditions”, Operations Research, 25(3), 517-524.

Saleh HA, Chelouah R. (2004), “The design of the global navigation satellite system surveying networks using genetic algorithms”, Engineering Applications of Artifical Intelligence, 17, 111-122.

Sofge D, Schultz A, De Jong K. (2002), “Evolutionary computational approaches to solving the multiple traveling salesman problem using a neighborhood attractor schema” Lecture notes in computer science, vol. 2279, Berlin, Springer, p, 151 – 160.

Somhom S, Modares A, Enkawa T. (1999), “Competition-based neural network for the multiple traveling salesmen problem with minmax objective”, Computers and Operations Research, 26(4), 395 – 407.

Tavakkoli Moghaddama, M.B. Aryanezhadb, N.SafaeibA. Azaronc (2005) ” Solving a dynamic cell formation problem using metaheuristics” Applied Mathematics and Computation Volume 170, Issue 2, 15 November 2005, Pages 761-780

Song C, Lee K, Lee WD.(2003), “Extended simulated annealing for augmented TSP and multi-salesmen TSP”, Proceedings of the international joint conference on neural networks, vol. 3, p, 2340 – 2343.

Srinivas, N., Deb, K., (1994), "Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. Evolutionary Computation", Wiley, New York.

Starkweather T, McDaniel S, Mathias K, Whitley D.(1991), “A comparison of genetic sequencing operators”, Proc. Fourth Int.Conf Genetic Algorithms and their Applications (Edited by R. K. Belew), pp, 69-76, Kaufmann.

Svestka JA, Huckfeldt VE. (1973), "Computational experience with an m-salesman traveling salesman algorithm", Management Science, 19(7), 790-799.

Tang L, Liu J, Rong A, Yang Z.(2000), "A multiple traveling salesman problem model for hot rolling scheduling in Shangai Baoshan Iron & Steel Complex", European Journal of Operational Research, 124, 267 – 82.

Toth P, Vigo D. (2002), "Branch-and-bound algorithms for the capacitated VRP", SIAM Monographs on discrete Mathematics and Applications, Philadelphia, p, 29-51.

Vakhutinsky IA, Golden LB. (1994), "Solving vehicle routing problems using elastic net", Proceedings of the IEEE international conference on neural network, p, 4535 – 4540.

Wacholder E, Han J, Mann RC. (1989), "A neural network algorithm for the multiple traveling salesmen problem". Biology in Cybernetics, 61, 11 – 19.

Wang X, Regan AC. (2002), "Local truckload pickup and delivery with hard time window constraints", Transportation Research, Part B, 36, 97-112.

Whitley D, Starkweather T, Fuquay D. (1989), "Scheduling problems and traveling salesman: the genetic edge recombination and operator", Proc. Int. Conf. Genetic Algorithms and their Applications, pp, 133-140, Kaufmann.

Yu Z, Jinhai L, Guochang G, Rubo Z, Haiyan Y. (2002), "An implementation of evolutionary computation for path planning of cooperative mobile robots", Proceedings of the fourth world congress on intelligent control and automation, vol.3.

Zhang T, GruverWA, Smith MH. (1999), "Team scheduling by genetic search", Proceedings of the second international conference on intelligent processing and manufacturing of materials, vol.2, p. 839-844.